

### **Amendments to the Specification**

Please replace paragraph [0027] using paragraph numbers as in the published application with the following amended paragraph:

[0027] In FIG. 4, an example of just-in-time (JIT) dynamic emulation is represented where the native legacy code (LC) in object code form is translated (by translator 22 of FIG. 2), is cached (in cache [[22]] 23 of FIG. 2), and executed (in execution unit 13-F of FIG. 2) a small portion at a time. The translation occurs for only a small portion of guest object code in groups that are likely to be executed next. The translation is performed in real time in the host system and is essentially concurrent with the execution of the translated code. The translated and cached code can be subsequently re-used without the need for re-translation.

Please replace paragraph [0028] using paragraph numbers as in the published application with the following amended paragraph:

[0028] In FIG. 4, the example CISC code of FIG. 3 is executed using the tracking table [[23]] 22 and cache [[22]] 23 of FIG. 2 in a host system 16 of FIG. 1. In operation, the host system accesses the legacy code (LC) of FIG. 4 in groups where, as in FIG. 3, there are three 16-byte aligned blocks (a first block at addresses 100, 102, 106, 10C; a second block at addresses 110, 114, 118, 11A; and a third block at addresses 120, 128, 12A). Each CISC block is translated into a block of corresponding RISC code by translating each CISC instruction in a block in order. As in FIG. 3, one or more RISC instructions are required to perform the equivalent function of each CISC instruction depending on the degree of complexity of each CISC instruction.